

Change Log for MQX™ RTOS from Version 3.0.0 to 4.2.0

1 Introduction

This document contains an overview of feature changes between versions 4.2.0 and 3.0.0. The change log maps the history of MQX™ RTOS feature evolution and enables users to select an MQX RTOS version that best fits their needs.

Contents

2. Version 4.2.0 (May 2015)	2
3. Version 4.1.0 (Feb 2014)	4
4. Version 4.0.1 (May 2013)	6
5. Version 4.0.2 (Aug 2013)	7
6. Version 4.0.0 (Dec 2012)	8
7. Version 3.8.1 (Jun 2012)	12
8. Version 3.8.0 (Dec 2011)	14
9. Version 3.7.0 (April 2011)	19
10. Version 3.6.2 (Nov 2010)	22
11. Version 3.6.1 (Aug 2010)	23
12. Version 3.6.0 (Jun 2010)	25
13. Version 3.5.1 (Feb 2010)	30
14. Version 3.5.0 (Jan 2010)	30
15. Version 3.4.0 (Sep 2009)	33
16. Version 3.3.0 (Aug 2009)	34
17. Version 3.2.1 (May 2009)	35
18. Version 3.2.0 (May 2009)	36
19. Version 3.1.0 (April 2009)	36
20. Version 3.0.1 (Jan 2009)	37
21. Version 3.0.0 (Dec 2008)	38

2 Version 4.2.0 (May 2015)

- Following Board Support Packages were added:
 - FRDM-K22F
 - TWR-K21F120M
 - TWR-K22F120M
 - TWR-K65F180M
 - Vybrid Auto EVB board was replaced by new SVF522R-EVB board
- Build projects for all available development tools have been updated to support newest tool-chain revisions available at the time of MQX RTOS 4.2.0 release - see chapter Development Tools Requirements for full list of supported build tools.
- MQX RTOS Kernel Components - version 4.2.0
- Minor changes and bugfixes
- RTCS TCP/IPv4 and TCP/IPv6 (optional) stacks - version 4.2.0 (same as in KSDK 1.2.0)
 - The new TLSF best-fit memory allocators were added.
 - LLMNR - RTCS newly support Link-Local Multicast Name Resolution (LLMNR) Server. This protocol allows resolving simple label names on local subnet without the necessity of having a DNS server.
 - Added DHCP Client v6 client application protocol support (Available in MQX RTOS IPv6 Add-on for purchase).
 - Added Telnet Client IPv6 protocol support; API of this component was changed (Available in MQX RTOS IPv6 Add-on for purchase).
 - Added TFTP Client/Server IPv6 protocol support; API of this component was changed (Available in MQX IPv6 Add-on for purchase).
 - Added support of WebSocket server as part of Http server code. The implementation was fully tested by AutoBahn test suite.
 - HTTP server was extended by SSL support (Available in WolfSSL add-on for evaluation).
 - Socket code has been updated by various BSD compatible options and flags.
 - ARP cache handling was modified to protect RTCS against denial-of-service attack.
 - For more details see complete change log in the <mqx>/rtcs/rtcs_changelog.txt file.
- MFS FAT file system - version 4.2.0 (same as in KSDK 1.2.0)
 - Support for multiple concurrent readers/writers of the same file
 - Support of mutex in addition of lwsem - solution to priority inversion
 - Locking mechanism reworked
 - Read only checks - improved efficiency
 - Better abstraction of FAT chain handling – better performance of random seek
 - Scalable sector cache layer – dedicated directory and FAT sector buffers were removed
 - True file append mode – seek to end of file atomically prior to each write operation
 - Directory search using FAT chain abstraction - improved efficiency

- Extension to find first/next API for direct long filename extraction - significant performance boost
- Path parsing redesigned to avoid unnecessary allocation of path buffers - lower RAM footprint
- Unicode support - UTF-8 is used to encode extracting and comparing existing filenames (note: UTF-8 fully not supported for writing)
- MQX RTOS Host and Device USB stack version 4.2.0
 - Legacy USB stack with minor changes and bugfixes. Includes support for legacy platforms only (same processors previously supported in MQX RTOS 4.1.1.)
- New Host and Device USB stack version 2 (same as in KSDK 1.2.0).
 - New USB stack is available for the following platforms : FRDM-K64F, FRDM-K22F, TWR-K21F120M, TWRK22F120M, TWR-K64F120M, TWR-K65F180M, TWR-VF65GS10, SVF522R EVB.
 - Support added for Composite device and PHDC class (Peripheral and Host)
- CyaSSL evaluation package (Available in optional WolfSSL add-on for evaluation) - version 3.3.0 (same as in KSDK 1.2.0)
 - The CyaSSL library support was updated to version 3.3.0 .
 - Resolved problem with memory leak in RTCS SSL wrapper. The memory resources were incorrectly deallocated if the underlying sockets report error conditions.
- IEEE 1588 Precise time protocol evaluation package (Available in optional IEEE 1588 add-on).
 - Newly supported RTCS extension based on IXXAT IEEE 1588 stack (time limited evaluation only).

Resolved issues:

- MQX-4593 ARP shell command does not work with ENET device number 1. Problem solved with added multi-interface support.
- MQX-4204 Lwtimer callback wasn't called when that task was blocked. The issue was fixed with setting default stdin, stdout and stderr for idle task in `_bsp_init()`. When printing from ISR is used, standard output is used from active task. Printing from ISR is now possible if the interrupted task is idle task.
- MQX-4470 MFS incorrect update of directory record was fixed.
- MQX-4406 Change of bool type size and casting return types can cause stack corruption. IOCTL bool param handling in `_io_cdc_serial_ioctl()` corrected. IOCTL bool param handling in `NANDFLASH_IOCTL_WRITE_PROTECTED` ioctl + minor changes in esai example have been made.
- MQX-4146 MFS – File size is not updated in file handler after successful writing on MFS device. Problem solved by unconditionally update of `SIZE` in `MQX_FILE_PTR`.
- MQX-4114 Function `_task_get_parameter_for` does not check the invalid input parameter. The issue was solved by added error check into function `_task_get_parameter_for()`.
- MQX-3915 FlexCan shutdown does not work correctly. The issue was solved by adding destroy lwevent in shutdown process, some functions to uninstall interrupt and disabling interrupt in function `flexcan_shutdown()`.

- MQX-3883 UART – Miss ioctl function IO_IOCTL_SERIAL_GET_CONFIG solved by added implementation of missing functionality.
- MQX-3887 I2C – Ioctl function still return OK when call it with invalid parameter. Solved by added IO_ERROR_INVALID_IOCTL_CMD return value for the wrong IOCTL command into driver.
- MQX-3881 UART – Miss the operation with “RX_NOISE_ERRORS” in UART statistical information was fixed.
- MQX-1110 RTCS/HttpServer – Wrong access of data in session buffer while searching for SSI was fixed.
- MQX-4387 RTCS- select() does not return activity if fd_set has zero socket handles. Solved with first non-zero nfds sockets in each fd_set are examined for activity.
- MQX-4265 RTCS - function getaddrinfo() fails if address is specified was fixed.
- MQX-4461 RTCS - Httpsrv chatclient disconnect with transferring a large amount of characters. Problem was resolved by ignoring data without ID instead of closing connection.
- MQX-4196 RTCS - C++ includes wrapper missing from MQX RTOS files. Resolved with _cplusplus condition to all RTCS header files.
- MQX-4019 TCP/IP – Unhandled Interrupt generated when using IPv6. The issue was solved with added missed parameter-structure deletion from the parameter-structure list, when ping6 timeout occurs.
- MQX-3984 RTCS – Bug in sending TCP packets causes dropped connections through firewalls. Problem was solved with unconditional sending MSS and window size with SYN.
- MQX-1353 RTCS/ICMP – ICMP redirect to invalid gateway is not rejected. The issue was solved with added additional check for routing address.
- MQX-1086 RTCS – The issue with TFTP client returning the wrong message if file is not found was resolved.

3 Version 4.1.0 (Feb 2014)

- New Board Support Package added
 - TWR-K21F120M
- Vybrid-based board support packages for TWR-VF65GS10 and AutoEVB updated
 - Support of audio-related drivers extended--SAI, eSAI, and ASRC. Device driver for DSP Codec CS422888 added (AutoEVB only).
 - Support the Quadrature Decoder functionality in the FTM peripheral
 - Added eDMA driver
 - UART driver updated to use the eDMA
 - Clock management component ported
 - Added DCU driver
 - QSPI driver updated to support the FlashX framework
 - NAND FFS support added

- DMA support in device drivers has been extended
 - Introduced new DMA device driver, supporting eDMA peripheral on Kinetis and Vybrid processor families
 - The SPI device driver was updated to support the new DMA driver
 - The SAI and eSAI audio drivers support DMA
 - eSDHC drivers were reworked to fully leverage the ADMA peripheral module
- Driver updates
 - The LWADC driver has been ported to all supported board support packages. The support of the legacy ADC driver was discontinued.
 - The RTC driver was updated on all supported platforms. Provided generic, POSIX compatible API for time conversion functionality.
 - FlashX driver extended by handling Flash Swap functionality on Kinetis processors.
 - LP Timer module was added to the HW Timer framework. Its usage is demonstrated in the Low Power and HW Timer example applications.
- Standardization effort
 - Legacy MQX RTOS custom integer types were replaced by the Standard C99 set (int_32 -> int32_t, boolean -> bool, etc). A header file is provided with the set of backward compatible type definitions to make the transition to the new types easier. For more details, see Section 3.1 “C99 Types” in the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS).
 - The endian conversion macros were consolidated inside MQX RTOS. The htons, ntons and similar conversion functions were renamed to mxq_htons, mxq_nton to avoid a conflict with the standard.
- NAND FFS library is no longer provided as a separate add in package but it is directly included as a part of MQX RTOS main package
- RTCS new features and enhancements
 - The MQX RTOS TCP/IP stack is now available with an optional package to enable the IPv6 protocol support. For more information visit (freescale.com/mqx)
 - The FTP server was redesigned to provide faster and more stable implementation.
 - The DNS resolver was updated.
- USB
 - Fixed several EHCI related bugs (HUB, pipe close, audio example)
- MQX RTOS startup is now split in two parts to avoid a crash risk if an interrupt occurs during the startup.
 - `_bsp_enable_card()` function has been replaced by the `_bsp_pre_init()` function that handles initialization of the OS vital functions, such as the timer (system tick), interrupt controller, memory management, etc. The `_bsp_pre_init()` function is called during the MQX RTOS initialization time, before any user task is created and the scheduler is not started.
 - The second part of the startup is done in a separate `_mqx_init_task` that executes `_bsp_init()` function for I/O drivers or stacks initialization and `_bsp_post_init()` function for possible

post-init operations. After the `_bsp_post_init()` function execution, the `_mqx_init_task` is destroyed.

- All BSPs are now adjusted to this concept. All I/O drivers are installed in the context of the `_mqx_init_task` after the MQX RTOS scheduler is started. This concept also allows a complex driver installation (handling ISRs during the driver initialization, drivers can use blocking functionality like `_time_delay`, etc.).

4 Version 4.0.1 (May 2013)

- New architecture ARM®Cortex®-A5 and new processor family Vybrid support added.
 - MQX kernel written in ARM ASM, C-sources compiled in thumb mode
 - NEON and FPU supported
 - The `mem_copy` function optimized for Cortex-A5 platform using NEON instruction set
 - L1 and L2 cache enabled
- New board support packages added:
 - TWR-VF65GS10_A5 – support for Cortex-A5 core for dual core Vybrid processor
 - TWR-VF65GS10_M4 – support for Cortex®-M4 core for dual core Vybrid processor
- MQX RTOS ported to new compilers and build tools:
 - Code Warrior 10.3 GCC build tools fully supported – IDE projects and makefiles provided
 - ARM Development Studio 5 (DS-5) – IDE projects provided (Vybrid only)
 - Sourcery CodeBench Lite Edition – makefiles provided (Vybrid only)
- Introduced new HW Timer driver – generic interface to various timer modules. It is currently in experimental phase and supports Kinetis and Vybrid PIT HW module only.
- Multicore Communication (MCC) API provided allowing inter core (OS) communication
- Tested with MQX RTOS to MQX RTOS, MQX RTOS to Linux® OS
- BSP Cloning wizard was updated
 - CW10.3 GCC support
 - New option to export whole MQX RTOS source tree for given board
- Extended IPv6 support - 85% of the “IPv6 Ready Logo” Phase-2 tests (Sections 1, 2, 3, and 5) passed.

RTCS Bugfixes

- Added "ipconfig <device> release" shell command.
- Added additional command line parameters (count, hop limit, interval, pattern, and size) for the ping shell function.
- Implemented missing interface release functions: `ipcfg_release_interface()`, `ipcfg_release_device()`.
- Fixed `ENET_shutdown()` not releasing all allocated resources.
- Calculation of time interval was replaced by `RTCS_timer_get_interval()` to avoid the time-rollover issue.

- Replaced IPs from Microsoft private pool (169.254.x.x) which were used in examples by more generic IPs (192.168.1.x).
- Fixed problems with TFTP PUT function.
- Fixed incorrect echo handling in telnet.
- Other minor fixes.

USB Bugfixes

- Fixed the IDT leak issue - the uncached memory allocation for each qTD/ ITD/ SITD was added to the code.
- Fixed USB Host ISO transfer buffer to be aligned to 4K boundary.
- Added missing inf files for both virtual nic and virtual com example applications.
- MFS Bugfixes
- Fixed - MFS rename functionality. The rename of file to another directory required filename to be the same.

MQX RTOS Bugfixes

- DSPI driver - last bit always 1 in SPI MODE1/3 - ASC field of CTAR register set to correspond with half period of SCK.
- Fixed FlashX driver access to the sector at the top of addressable space (i.e., at 0xffff_ffff).
- Fix in shell_write function to ensure proper reporting of errors when a file larger than the disk is written.
- Fixed fread(), zero length parameter read; behavior changed to match other implementations (Linux OS).
- The lwmsgq_send function now checks if it is running in ISR. Input blocking flags then cause return error MQX_CANNOT_CALL_FUNCTION_FROM_ISR.
- Fixed multiple PDB setup for ADC channels with different base frequency.
- Added extern "C" to nvic.h file.
- Fixed FlashX driver incorrectly handling file open/close which was leading to memory leak.
- FlashX driver - added NULL pointer checks to ftfl_init_ram_function() and ftfl_flash_init().
- Fixed cache flush/invalidate functionality when multiple lines for unaligned blocks of data are used.

5 Version 4.0.2 (Aug 2013)

- AutoEVB Vybrid ARM Cortex-A5 and Cortex-M4 Board support packages
 - Support for Cortex-A5 and Cortex-M4 cores of the dual core Vybrid processor
 - Supporting standard set of I/O drivers
- Vybrid BSP extensions
 - RTC and NAND flash driver ported to Vybrid platform
 - MMU support extended to handle 4 KB memory blocks
 - Introduced new QuadSPI driver

- Cortex-M4 boot option enabled.
- Direct code execution from QuadSPI flash – XIP provided. This feature is demonstrated in Vybrid QuadSPI bootloader.
- Introduced the first version of eDMA driver – the driver is experimental and will be further extended in the next MQX RTOS version. Documentation is not provided.
- Kinetis SPI driver was updated to use eDMA driver.
- HW timer driver support extended to SysTick and GPT HW modules. Kinetis and Vybrid BSPs updated to leverage HW timer as the provider for MQX RTOS tick time.
- RTCS HW checksum acceleration enabled in enet driver (for TCP and UDP); the benefits are increased throughput and reduced processor loading. Option enabled by default for K60N512 platform.
- HTTP server redesigned to provide a faster and more robust solution. The server API was simplified and changed to correspond with the RTCS standard.
- New SMTP client functionality provided as a part of RTCS network suite. The client offers a simple API for email handling. The new RTCS Shell command “email” demonstrates its functionality.
- The security_email, security_webserver, and security_telnet demo applications were removed from this release. Networking functionality is demonstrated in RTCS shell and httpsrv example applications.
- Multicore Communication (MCC) library updated to version 1.1. Incorrect usage of cache macros in mcc_send() function fixed.

6 Version 4.0.0 (Dec 2012)

- New board support packages added:
 - TWR-K21D50M
 - TWR-PXD10 - MQX RTOS kernel was ported to e200 VLE assembly instructions
- MQX RTOS ported to new compilers and build tools:
 - Makefiles for command-line GCC build provided for selected Kinetis BSPs (TWR-K60N512, TWR-K60D100M, TWR-K70F120M). See the makefile files in the /build/<board>/make folder.
- Ready for new Freescale Vybrid PSP and Cortex-A5 processor core support.
 - The MQX RTOS 4.0 has been tested with the Cortex-A PSP and Freescale Vybrid BSP libraries which are available as a standalone installation package.
- Ready for new Freescale NAND Flash File System library.
 - The MQX RTOS 4.0 has been tested with the new FFS library which implements the wear leveling algorithms and support for on-chip NAND Flash controller.
 - This library enables the MFS filesystem to be installed and used on top of the NAND Flash memory devices.
 - The FFS library is available as a standalone installation package.

- RTCS stack was extended by IPv6 protocol support, functionality provided was fully tested by legacy IPv6 Silver logo test set. The IPv6 support is available in the separate installation package. Currently supported protocols and services are:
 - IPv6 IP layer, ICMPv6/ping, HTTP server
- The following board support packages were removed from the release and will be supported only as a part of MQX RTOS 3.8.1 version maintenance.
 - TWR-MCF51AG, TWR-MCF51CN, TWR-MCF51JE, TWR-MCF51MM, TWR-MCF51QM, DEMOEM, DEMOAC, EVB51JM128, M5208EVB, M52223EVB, M52233DEMO, M52235EVB, M52277EVB, M53015EVB, M5329EVB, M54455EVB and MPC8308RDB
- Consolidated source files with generic kernel implementation. Number of PSP files was reduced which results in faster library build time.
- Support of CodeWarrior Classic development environment was dropped and all library and example projects were updated to CodeWarrior 10.2.
- Pre-compiled binary libraries are no longer included in the installer. For detailed build instructions please see “Building the MQX RTOS Libraries” section in the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRDOS) document.
- The MFS read and write speed was improved leveraging the multi-sector transfer functionality and reworked SPI and SDHC drivers. Measured maximum speed is up to 10 times higher when 32 K blocks are used (~10MB read/2,5 MB write for class 10 SD Card and TWRK40D100@96 Mhz board)
- RTCS TCP/IP code has been updated and throughput has been improved.
 - Measured by Freescale FNET fBench tool: TCP TX~11Mbps; TCP RX~5Mbps; UDP TX ~25Mbps; UDP RX~24Mbps tested with TWRK60D100@96Mhz board at default configuration.
- New I2S & SAI audio drivers added for Kinetis processor family. Example application is provided.
- USB Audio examples updated – microphone and speaker examples are available for selected platforms.

RTCS Bugfixes

- The dual_webserver example application was removed from release and current webserver application has been extended to handle two and more Ethernet interfaces instead
- FTP long file names can cause stack overwrite –file name handling corrected.
- Several PPP bugs resolved during the PPP code update (PPP PAP authentication, shutdown, reconnect etc.). New RTCS shell commands for PPP handling were introduced.
- A return value of the pseudo_phy_discover_addr() in the pseudo_phy.c source file was added.
- The HTTP server in static task setting had the hard coded priority - the static task priority value is derived from HTTPDCFG_DEF_SERVER_PRIO macro now.
- The HTTP server was extended by "keep-alive" functionality.
- The RTCS_selectall() documentation was modified to match intended functionality – the RTCS_selectall() waits on all sockets in the system

- DNS problem was caused by disabled loopback interface in the default RTCS configuration. Fixed by proper configuration in the rtscfg.h file.

Build Tools related bugfixes

- Several changes were made in the CodeWarrior projects to be compatible with the upcoming 10.3 version. For example, use of slash in the configuration name has been removed and more compatibility changes done.
- Missing nandflash example was added to the New Project Wizard
- Performance Tool documentation link was fixed and documentation was updated.
- Invalid project paths referencing the PE_LDD macros were removed from the PE-enabled projects.
- Fixed C++ operators new/delete to be working correctly in Keil IDE. Consolidated overriding of malloc, calloc and free functions to use memory allocation system of MQX RTOS for all compilers.

MFS / IO subsystem bugfixes

- Attempt to delete open file is correctly denied with MFS_SHARING_VIOLATION.
- The _io_mfs_dir_read() API time and date handling corrected. Added check for month value, if the value is not between valid range (from 1 to 12), the "???" string is printed out instead of month abbreviation.
- The fseek/fwrite functionality was fixed to work according to standard if the fseek() moves position beyond the EOF (the file is extended by zeros to the fseek position upon the following fwrite).
- Functionality of _sem_wait_ticks fixed on ColdFire processor. The functions did not waited for expected number of ticks.
- The printf() function fixed to correctly print out the zero character and also other formatting now matches the GCC standard style.
- Corrected printing of different sized variables on big endian platforms (PPC, ColdFire)
- Shell dir command has been modified to correctly display directory content for file systems which are not current.

USB

- Both USB Host and Device stacks were significantly re-factored and reviewed. Compatibility issues of USB Mass Storage devices which were reported for older versions were reviewed and fixed.
- USB audio examples were reworked and updated to use I2S audio driver and SD card wav file read/write.
- Fixed problem of USB Host Pipe structure list being corrupted in some circumstances when USB device was re-attached.
- The EHCI Pipe private data were not cleared properly when EHCI pipe got free.
- The USB Host stack HighSpeed HUB functionality was fixed

Example application modifications and bugfixes

- The mutex example was changed to better demonstrate the mutex features

- Incorrect usage of feof() in hvac demos was corrected. Added condition checking for both EOF and error code to the example application.
- The mfs_usb example was not de-allocating all resources from the heap when USB stick was unplugged. The memory leak fixed in the example application.
- The ADC potentiometer channel macro fixed in BSPs and is now used consistently in example applications. The macro was fixed to use single-ended mode on the following platforms TWR-K60F120M, TWR-K70F120M and TWR-K53N51.

Other bugfixes

- Kinetis ADC driver was updated: ADC_MAX_SW_CHANNELS was modified, adc_max_frq_table[] limit values were corrected to 12MHz at 16bit resolution and to 18MHz for lower resolution modes.
- TWR-PXN20 BSP FlashX driver configuration (block size and with) was corrected to match the processor flash specification.
- The FlashX and FlexNVM functionality has been implemented for TWR-MCF51JF and TWR-K40D100M.
- Kinetis FlashX driver updated. Fixed incorrect functionality if the number of sectors was not divisible by 8. Fixed chip erase IOCTL command. Corrected problem in io_flashx_find_correct_sectors() if a flashx_file spreads over more than one flashx_block.
- Several ADC driver fixes made on Kinetis platform : Accumulate functionality was modified, the ADC_VALUE_TYPE changed to 32 bits, the GPIO init function for TWR-K60F100M and TWR-K70F100M corrected.
- The LWGPIO driver was fixed to correctly handle PORTTH pins on MCF52259 processor.
- The functionality of _mem_alloc_at() function was fixed. The issue appeared when this API was used more than once in the application code.
- The _task_disable_fp(), _task_enable_fp() API functions were implemented for Kinetis architecture.
- Fixed race condition in the _lwmem_alloc_internal() which can lead to the incorrect memory allocation in some circumstances.
- Changed size of FLASH memory from 512kB to 1MB in linker command files for TWR-K60F120M board. Changed initialization struct for FlashX driver from 2 banks to 4 banks.
- RAM vectors table feature was fixed on all Kinetis platforms for ARM Keil Vision IDE.
- The BSS section definition was fixed in the cw10 linker command file for twrk60n512. The .bss and .sbss (zero initialized data) was located incorrectly in the main_application_data section resulting in larger flash footprint.
- The function nfc_write_page() did not return NANDFLASHERR_TIMEOUT unless the time out occurs for all pages being written. The behavior was corrected to be able to detect the error on any page being written.
- Size of ROM vector table has been corrected from 0x410 to 0x400 in the linker command file for CodeWarrior 10.2 and TWR-K60F120M BSP.

7 Version 3.8.1 (Jun 2012)

- New Board Support Packages added
 - TWR-K20D50M
 - TWR-K20D72M
 - TWR-K40D100M
 - TWR-K60D100M
 - TWR-K60F120M
 - TWR-PXS30
 - TWR-PXN20
- Introduced new LWADC driver implemented for PX processor family. Documentation updated, new example application `mqx/examples/lwadc` has been prepared
- TWR-K70F120M BSP update
 - System cache has been enabled, resulting multiply time higher speed in IntFlash DDR data targets
 - USB High Speed (EHCI) operation has been enabled
 - Processor Expert support has been added (allowing use of PE drivers and Low Power features)
- TWR-PXS20 BSP update
 - Added LW ADC driver
- USB Host library has been updated to be able to handle both Full Speed (EHCI) and High Speed (KHCI) modules simultaneously.
- User mode support extended to Keil and CW10 toolsets for K60N512 BSP
- The Low Power features and Processor Expert support have been extended to TWR-MCF51JF and TWR-MCF51QM BSPs
- Sleep on Idle – MQX RTOS implements possibility to put processor core to the sleep mode and save energy during execution of the idle task. This feature is currently available for ColdFire and Kinetis platforms only. See details in *Freescall MQX™ RTOS I/O Drivers User's Guide* (document MQXIOUG) Chapter Low Power Manager.
- Low Power Timer driver has been implemented and the functionality has been demonstrated in the `mqx/example/lowpower` application.
- SDCARD and DSPI and SDHC driver improvements
 - The clock speed of setting of communication channel (SPI, SDHC) has been corrected
 - DSPI polled driver optimized for performance, increased throughput at higher baud rates
 - CRC calculation in `sdcard_spi` driver implemented using a table, significant performance increase
 - SDCARD drivers prepared to make use of multi-sector transfers in the future (next release)
 - ESDHC driver does not use intermediate buffer for data freeing up 0.5 KB of RAM

Bugfixes:

- MPC5125 serial with interrupt is not working – serial driver `putc` routine has been fixed

- I2C driver on mcf51jm gets stuck – prepared workaround using short wait loops before reading of IIC status register
- Compiler warnings for CW10.2 - #pragma opt_branchtailmerge off – removed obsolete compiler macro
- User mode tests fail in the INTRAM targets – linker command file sections has been corrected
- NFC Driver - incorrect ECC status word reading - ECC status word reading fixed
- SPI example application failed for TWRPXS20 BSP chip select of connected accelerometer was disabled by pin pull-up resistor and rx buffer overflow was prevented in interrupt driver ISR
- MQX RTOS Telnet server update – incorrect ECHO handling
- Support negotiation of binary mode.
- Introduced new modes CR LF, CR NULL and character block binary mode.
- Introduced new IOCTL commands to switch between modes.
- Update demos to handle character mapping between application and telnet NVT.
- RTC for mcf5xxx - date gets out of valid range due to wrong RTC_BASE_YEAR
- SPI clock line is not set up during spi_init TWR-MPC5125 BSP– BSP GPIO setting fixed, added missing IOCTL commands
- Printf for floating types compatibility issue - Differencies between gnu libs and MQX RTOS representation was fixed.
- ipcfg_bind_staticip do not fail if the same address is present on network - IPIF_bind() corrected
- The C library exit() function call was removed from MQX RTOS all example applications (replaced by _task_block()) to allow stdout output channel to be flushed properly and also to enable easier debugging of example applications if errors occur.
- In the full memory allocator, the highwater mark was updated incorrectly when an allocation was made in the pool extension area (created by _mem_extend()) - Now the highwater mark is updated only if block is allocated from the main pool area - so it shows valid information. The highwater mark is not available for pool extension areas.
- The TAD was fixed and updated to display all allocated memory blocks, even the ones from the memory extension areas created by _mem_extend() call. In the previous implementation, the TAD might have failed to display the memory blocks correctly if the extension area was created.
- Linker Command File TWRMPC5125 modified - kernel_data start symbol is placed automatically after statically allocated data now.
- PAP authentication fixed for PPP connection - problem has been fixed in both PAP&CHAP state machine and example application. See example readme, pap.c and chap.c files for details.
- Functionality for length modifiers (h hh and L ll) was implemented for printf(), fprintf(), sprintf(), snprintf() functions.
- No source available while debugging the TWR-PXS20 platform - All example project for have been updated with correct setting (“Generate DWARF information”)
- Solved problem with reattaching the low speed devices to MQX RTOS USB Host. The attach failed every second time because of incorrect speed handshake algorithm
- Data and instruction cache has been enabled for TWRK70F120M board in default setting

- TWRK70F120M – linker command files for intflash_dddrrdata and intflash_sramdata have been fixed – flash (ROM) space is fully used now
- IPC UDP example was not working properly. IPC used default initialization structure because no custom one was passed. '_task_create' call was modified and it now correctly passes this structure to IPC initialization task.
- Bug if the Kinetis ADC fopen/fclose was called multiple times fixed in PDB initialization.
- Flashx driver on twrmcf51mm and twrmcf51je supports now 2 banks of internal flash memory.
- Kinetis serial driver now allows to set 1 or 2 stop bits using IOCTL command IO_IOCTL_SERIAL_SET_STOP_BITS. Since the HW doesn't support variable number of stop bits, the second stop bit is emulated using 9bit communication with last bit set to 1. The 2 stop bits configuration is therefore available only in 8 bit data mode with parity disabled.
- MQX-memcpy() failed, when KEIL-optimization is enabled – mem_copy functionality corrected and by default the MQX RTOS uses use runtime (tool provided) memcpy instead of mqx_mem_copy to achieve best performance
- MFS_Get_disk_free_space() - fixed free space calculation in MFS
- When ping request expires due to timeout, the associated ARP request queue is being cleared. The clearing code was fixed so it doesn't block RTCS task if entries in the queue don't match the ping request.
- IODEBUG documentation has been added to *Freescale MQX™ RTOS I/O Drivers User's Guide* (document MQXIOUG)

8 Version 3.8.0 (Dec 2011)

- New Board Support Packages added
 - TWR-K70F120M
 - TWR-K53N512
 - KwikStik (K40X256)
 - TWR-MCF51JF
 - TWR-MCF51QM
 - MPC8308-RDB
 - TWR-PXS20
- MQX RTOS, IO, RTCS, MFS and USB documentation updated. New USB User Guide documents created.
- New documents prepared to describe tool-specific settings for TAD and other MQX RTOS debugger plug-ins for CodeWarrior 10.x, IAR EWARM, ARM MDK - Keil Vision environments.
- The Kinetis port of the Freescale MQX RTOS is newly supported by the ARM MDK – Keil Vision IDE. The support includes:
 - Projects files for standard set of libraries: PSP, BSP, RTCS, MFS, USB Host and Device.
 - The uvmpw workspace files ready to be used to batch-build all MQX RTOS libraries.
 - Project files for all example applications and demo projects. Projects are tested with on-board P&E Micro OSJTAG interface and external ULINK® interface.

- All projects are prepared for Vision 4.22a and are set up for OSBDM debugging interface. The debugging connection may be changed to ULINK debugging interface – see more information in the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS).
- MQX RTOS Task Aware Debugging plug-in.
- The Cortex-M PSP (Kinetis kernel) has been modified to be able to create tasks running in user (restricted) mode. The lightweight memory management, dynamic task creation, lightweight semaphore, event and message API functions were implemented to operate from user mode. This is an experimental prototype of the feature available for K60 and IAR build tools only in this version.
- HW floating point support added to Kinetis PSP. This feature is supported on the new MK70F120M platform.
- New LWGPIO Driver - smaller and faster GPIO driver was implemented for all supported platforms. MQX RTOS example applications were updated to use this driver. The support of legacy GPIO driver is discontinued and it will not be ported to future platforms.
- New IODbg IO driver was created. This driver allows redirecting I/O functions (e.g., printf) to a debug probe communication channel. The CodeWarrior 10, IAR EWARM 6 or Keil Vision4 debugger consoles are supported. See the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS) for details on how to setup and use this feature.
- The FlashX driver and example application were rewritten to achieve effective operation and support of non-continuous Flash memory space used on Kinetis processors.
- I2C Slave driver functionality was re-implemented and new example applications have been created for all supported processors.
- TCHRES (Resistive touch screen) driver and example application demonstration for TWR-LCD board.
- The Ethernet bootloader example application has been created – this example application provides functionality similar to the well-known uBoot application. Only high-end platforms with external RAM are supported. Example contains implementation for both NAND and NOR flash memory devices.
- New C++ example application was created and C++ functionality was enabled on all platforms.
- Example project wizard introduced – users are able to create MQX RTOS example application projects directly from the CW10.x IDE.
- Added preliminary support for CodeWarrior 10.2 Beta. This requires manual installation of the CodeWarrior 10.2 plug-ins; see the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS) for more details.
- Created Working Set Importer plug-in in the CW 10.2. It allows developers to export and import Working Sets to/from XML file. The aim of this plug-in is to let the developer to import whole set of MQX RTOS libraries required for a chosen board at once. This feature is not supported for CW10.1.
- USB Host and Device code have been modified to support both little- and big-endian USB peripheral module.

- IAR EWARM 6: Assembler-coded routines have been annotated by special CFI directives. This enables the call-stack history to be displayed properly in the IAR IDE. Use the latest EWARM TAD plugin from <MQX install dir>/tools/iar_extensions/Embedded Workbench 6.X.
- Kinetis register description header files have been updated to reflect the latest Reference Manual.

Bugfixes:

- HTTP Server web pages which contains CGI does not work correctly in Chrome and Android™ browsers - Root cause for this issue was in inappropriate formatting of HTTP responses for CGI scripts and wrong Content-Length header field when the content is created dynamically.
- The `_mqx_exit()` call was eliminated from all examples (replaced by `_task_block()`) to allow stdout output channel to be flushed properly and also to enable easier debugging of example applications if errors occur.
- TFTP client cannot download file from TFTP server – incorrect use of `read()` fixed in the tftp shell command.
- MQX RTOS crash under ping-flood stress test of RTCS applications – Issue was caused by incorrect Ethernet TX and RX priority setting. The interrupt priority setting has been corrected (Rx and Tx interrupts need to be set at the same level) in all BSPs.
- RTC applications build fail for MPC8308.
- Task destroy does not clean lwmem-allocated blocks memory - the lwmem blocks are now freed during the owners task destroy.
- Documentation updated for `ENET_initialize()`.
- MTIM timer does not run on MCF51JF and MCF51QM - new MTIM16 driver implemented for these platforms.
- Fixed GPIO IRQ functionality with on DEMOEM board.
- MQX RTOS uses old Kinetis header files - all Kinetis header files has been updated to new file revisions.
- MPC8308RDB - missing `ipsum.cw` in installer package added.
- Log application build fail for Kwikstikk40x256.
- I2C interrupt priority levels on Kinetis corrected - set to 5, sublevels removed.
- When selecting second alternative interface on the USB device from MQX RTOS USB host, the host stack freezes on searching pipe handles for that interface.
- SPI timing issue in DSPI driver - DSPI driver now calculates appropriate delays corresponding to the selected baud rate.
- The UART2 channel setting on the TWR-MPC5125 was fixed.
- Serial driver `IO_IOCTL_SERIAL_GET_STOP_BITS` missing - functionality added to serial driver for Kinetis K40, K53, K60 and Coldfire MCF51QM, MCF51JF
- MQX RTOS does not check all registered DNS servers - fixed.
- `snprintf` function was fixed.
- Multiple initialization of DNS - `ipcfg_add_dns_ip()` has been removed.
- Race condition in `_task_get_id_from_name` and `_task_get_template_index` has been fixed.

- MPC8308 output libraries file names are incorrect – projects were re-generated with the correct name.
- Telnet performance increased by IOSOCK buffering.
- The RTCS telnet echo parameter option has been implemented.
- The MPC5125 MQX RTOS decrementer initialization bug fixed.
- Incorrect pin handling in the legacy MCF5225x GPIO driver was fixed.
- Incorrect Target Settings of the MPC8308 projects fixed.
- Internal implementation of taskat function fixed.
- MPC5125 MQX RTOS FIFO corrected
- MPC5125 CodeWarrior v 9.2 New Project Stationeries were fixed
- Printf and scanf support for 64-bit numbers and %L modifier.
- Fixed printf, sscanf and scanf functions not working when using floats (%f) on Kinetis.
- Floating point libraries setting fixed for MCF52277 platform.
- MQX RTOS does not support %g %G printf() format specifiers – the missing functionality was implemented.
- ADC driver for Kinetis does not handle some input channels correctly.
- Bug in MCF51AC ADC driver provides invalid results.
- Kinetis linker map file did not make use of 32 Kbytes RAM.
- Problem in reading free space in USB stick.
- K60 pin muxing in “init_gpio.c” for UART4 (ttye).
- TWRK60x512 cannot attach USB Keyboard.
- MPC5125 SD card driver cannot read 16GB cards.
- ENET_shutdown() does not clean up all resources.
- Bug in RTCS select() if timeout is less than 1 millisecond.
- RTCS tick timer event scheduling fixed.
- MCF54455 RMII setting fixed.
- Interrupts pending from a bootloader before MQX RTOS starts may cause unhandled exception.
- WebHVAC application crashes on TWR-K60N512 board – the issue partially resolved by disabling Flash Cache - erratum e2647 for early silicon revisions (mask 0M33Z, see known issues for details).
- WebHVAC sporadic erratic behavior observed also with Flash Cache disabled on the early K60N512 silicon revisions (mask 0M33Z) when using IAR EWARM tools. The linker file was changed to put code to smaller addresses and constant data to higher addresses in the Flash. Application crashes not observed anymore after the change.
- Incorrect initialization of I2C2 and I2C3 peripheral drivers fixed on TWR-MCF51JF and TWR-MCF51QM.
- Issues with writing large file to USB memory stick on Cayman - fixed.
- PHDC weighscale application bug - missing reset state handler in phd_state_mc_func functions array - fixed.

- `_time_ticks_to_xdate()` produces incorrect values - fixed.
- The `ip.c` file does not compile when `RTCS_CFG_ENABLE_NAT` is enabled - fixed.
- Documentation of lightweight object destruction corrected.
- Nested comment in 5329 code has been removed.
- Documentation corrected – removed features which are not supported by current release (`IO_IOCTL_SERIAL_START_BREAK` and `IO_IOCTL_SERIAL_STOP_BREAK`).
- MPC5125, setting invalid MQX RTOS time when RTC is enabled - Add check of `_time_from_date` function return value.
- TAD in IAR EWARM 6.21.4 freezes - LWMEM pools TAD screen has been fixed.
- TWR-MCF5418 demo `web_hvac` cannot browse webpages from USB - Problem with memory cache. Changed `MFS_mem_alloc_system_zero` function - replaced `_mem_alloc_system_zero` with `_mem_alloc_system_zero_uncached`.
- `_task_get_td()` returning NULL when using `SYSTEM_TASK_ID` as parameter - fixed.
- Legacy GPIO driver interrupts handling corrected on MCF51 platforms.
- Bootloader example application and documentation has been updated. Line buffers were not handled correctly (function `bootstrap_ddr_to_nand(void)`)
- Serial port driver `IO_SERIAL_HW_FLOW_CONTROL` I/O control command fixed for interrupt mode on MCF5225x and Kinetis platforms
- USB CDC host `fread()` functionality fixed. The bug appeared when the application tried to read less data than received. This bug caused freezing of the CDC IN pipe.
- MCF54418 RTC driver fixed – Improved register unlock procedure is used in `bsp_rtc_io_init()` to ensure that RTC is correctly set.
- The TWRMPC5125 timing was inaccurate (33,333 MHz oscillator vs. 32,768 MHz). Fixed the frequency values in the `lcf` files.
- DSPI baudrate calculation code modified - Baud rate calculation was extended by DBR setting to provide wider set of frequencies. Change is relevant for Kinetis and ColdFire processor families.
- All Kwikstikk40x256 host examples removed from release package - Kwikstik supports USB Device only.
- The MPC8308RDB I2C fixed - onboard I2C EEPROM uses 2 byte-wide memory addressing. I2C example was updated.
- TFTP client compatibility improved - TFTP read operation should ACK packets with already passed IDs and drop those with future IDs for better compatibility with some TFTP servers (www.dhcpserver.de). Timeout prolonged to 5 sec.
- Fixed PPP over UART client authentication code.
- Function `ungetc()` did not behave as expected. The `ungetc` behavior has been re-implemented. Other functions like `ftell`, `fread`, `fgetc`, `fflush` and `fseek` were made `ungetc` aware.
- Kinetis Int RAM Targets with vector table in RAM do not work correctly – problem fixed in IAR linker command file.
- Printf formatting correction - The hex formatting (`%Nx`) does not print leading zeros for integers - fixed.

- Incorrect usage of MQX_USE_SMALL_MEM_ZERO macro in _mem_zero() function for PowerPC can lead to the non optimal code generation.
- SPI baudrate was incorrect on MPC5125 due to incorrect endianness of baudrate divisor value in CCR register - fixed.

9 Version 3.7.0 (April 2011)

- New Board support packages:
 - MCF51AG-based TWR-MCF51AG BSP
 - MCF51JE-based TWR-MCF51JE BSP
 - MCF5208-based M5208EVB BSP
 - MCF5329-based M5329EVB BSP
 - MCF54418-based TWR-MCF54418 BSP
- Kinetis BSPs TWRK60X512 and TWRK40N256 modified:
 - Clock speed changed from 48MHz to 96MHz
 - Programmable Gain Amplifier – PGA enabled in ADC driver
 - Flash driver extended by FlexNVM functionality (EEPROM emulation)
- TWR-MPC5125 BSP was extended by:
 - USB Host and Device
 - NAND, SDHC, SD Card, SPI and I2C drivers
 - NAND boot targets and Nand Flashing targets are available. See details in *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS) Chapter TWR-MPC5125-KIT BSP.
- CodeWarrior Development Studio for Microcontrollers Version 10.1 support
 - TAD and New project wizard for Coldfire and Kinetis Platforms
 - MQX RTOS libraries, example and demo projects
- Kinetis BSP projects are CodeWarrior 10.1 Processor Expert Ready
 - Processor Expert drivers are enabled in MQX RTOS environment
 - Two BSPs with Processor Expert drivers enabled <mqx>/build/twrk40n256_pe and <mqx>/build/twrk60x513_pe
 - Example application demonstrating PE functionality /demo/pe_demo
 - Processor Expert drivers are supported for Kinetis platform only
- Kinetis scheduler enhanced
 - The _int_install_kernel_isr() call to install a non-MQX RTOS interrupt service routine into interrupt table is supported
 - The RAM-based vector table is supported.
- New smaller and faster version of GPIO driver prepared – LWGPIO driver. The driver is currently available for MCF52259 and MK40 and MK60 based BSPs only. See *Freescale MQX™ RTOS I/O Drivers User's Guide* (document MQXIOUG) doc/MQXIOUG.pdf documentation and

/mqx/examples/lwgpio example application for details. The driver will be ported to other platforms in the future MQX RTOS releases.

- Serial driver was extended by new options allowing RS485 half duplex communication. New ioctl command for handling HW flow control (RTS/CTS signal) and SCI parity were added.
- The output directory for MQX RTOS PSP and BSP libraries has been changed – instead of lib/<board_name>/mqx are newly used lib/<board_name>/psp and lib/<board_name>/bsp folders. See *Freescal MQX™ RTOS BSP Porting Guide* (document MQXBSPPG) doc/FSL_MQX_Porting_Guide.pdf for instructions if you are porting the project based on older MQX RTOS releases.
- Mem_copy and mem_zero functions were set to speed optimized version by default. The code is longer but up to seven times faster than original implementation.
- Byte-swapping macros optimized and unified in PSP library (psp.h)
- Handling of multicast packets has been simplified in RTCS, no functional change.
- Task priorities changed from 5 to 8 in several example applications. Note that task priorities below 7 have special functionality with regards to interrupt masking.
- HTTP server reworked. The server is able to operate in one of three modes: polled mode, static task pool or dynamic task created per session. See RTCS documentation for details.
- MFS USB example application modified to support multiple USB sticks simultaneously
- USB Audio Class support added into USB Host stack for KHCI based platforms. Example application prepared for ColdFire V2 and Kinetis based platforms (usb/host/examples/audio/audio_stream)
- TCPIP stack (RTCS) is newly available for platforms without Ethernet controller. Ready for WiFi and PPP support.
- CodeSize script and reports updated. The script is able to generate detailed reports for PSP, BSP and newly also RTCS, MFS and USB libraries
- User documentation was updated

Bugfixes

- Management of free memory for Kinetis devices using _mem_alloc_at or _lwmmem_alloc_at allocation has been fixed. Using _mem_alloc_at can lead to losing part of free memory.
- USB Virtual_com example code was fixed. Communication buffers are now allocated in un-cached memory.
- Serial driver: The IO_IOCTL_SERIAL_TRANSMIT_DONE command was corrected. The IO_IOCTL_SERIAL_SET_HW_SIGNAL and IO_IOCTL_SERIAL_SET_PARITY commands were implemented for all platforms
- SDHC sporadic read errors problem solved for Kinetis and MCF54418 devices. Default baud-rate has been reduced. Adaptive baudrate setting will be implemented in the future MQX RTOS released.
- The MCF54418 GPIO driver interrupt related functionality has been fixed.
- The NAND ECC and back block byte handling in the MCF54418 NAND flash driver has been fixed. Wrong ECC size was defined in BSP header file.

- Several USB Host stack EHCI bug-fixes improving stack stability and compatibility with same mass storage devices (Error handling during enumeration improved, improved packet receiving algorithm)
- MFS free cluster counting has been fixed.
- ROM section setting corrected in the linker files for MCF51CN, MCF51AC and MCF51JM BSPs.
- MFSCFG_FAT_CACHE_SIZE set to value 2 for all configurations to improve USB Mass Storage write speed for devices with EHCI peripheral module.
- IO_IOCTL_SERIAL_TRANSMIT_DONE result was inverted bug was fixed
- Register definition file MCF54455 platform was corrected (mcf5445_edma_struct)
- QSPI baud rate calculation source code was unified across all platforms. The QWR setting was corrected.
- USB EHCI Host does not handle data properly, the packets are sometime rejected – fixed in asynchronous scheduler; the code accessing overlay area was removed
- M53015EVB board does not start without debugger – fixed in BSP initialization code for this platform
- psp_prv.s file kernel logging related compilation check was corrected
- HVAC and Security Webserver demo applications was updated to run on Kinetis platforms and IAR compiler
- SNMP compilation macros have been fixed. The RTCSCFG_ENABLE_SNMP macro was added to all SNMP related files.
- MPC5125 RTC example build error has been resolved
- LWDNS behavior was corrected for TWRK60X512 platform. Endian swapping macro was incorrectly used.
- GPIO driver – fclose() on Kinetis based platforms was corrected. The function incorrectly uninstalled the interrupts for other open pins.
- USB Host stack enumeration error handling has been improved. The stack is trying to re-assert transaction if error occurs.
- USB Host stack reviewed to correctly handle Big to Little endian conversion. USB MSD device class endianness problem fixed on Kinetis platform.
- ADC driver for ColdFire V1 platforms was corrected – wrong use of lwevent in conversion complete lead to incorrect signaling of the end of measurement
- Compilation errors if RTCS_ENABLE_IP_REASSEMBLY was enabled were corrected
- The file lwe_gets.c (lw event get signaled) was not included in PSP libraries for all platforms
- Function FLEXCAN_Rx_message() returned FLEXCAN_NO_MESSAGE when FlexCAN message box overrun. The function has been changed to return FLEXCAN_MESSAGE_OVERRITTEN.

10 Version 3.6.2 (Nov 2010)

- EAR v2 version of the Freescale Kinetis ARM Cortex-M4 Platform Support Package was prepared. See known issues related to this version. The Kinetis port of the Freescale MQX RTOS is supported by the IAR Embedded Workbench for ARM. The support includes:
 - The EWARM workspace files ready to be used to batch-build all MQX RTOS libraries.
 - Projects files for standard set of libraries: PSP, BSP, RTCS, MFS, USB Host and Device.
 - Project files for all example applications and demo projects. Projects are tested with on-board P&E Micro OSJTAG interface and external Segger/IAR j-link interface.
 - By default all projects are prepared for EWARM 5.50.6 and are configured for IAR j-link debugging interface. The projects are ready to be converted to EWARM 6.10 format just by opening them in the new environment. In EWARM 6.10, the debugging connection may be changed to P&E Micro OSJTAG.
 - MQX RTOS Task Aware Debugging plug-in for both EWARM 5.50 and EWARM 6.10 versions.
- MK40X256-based TWR-K40X256 Board Support Package was added
 - Standard set of peripheral drivers is enabled: GPIO, I2C, INT_CTRL, UART, SPI, RTC, ADC, SD card and FlashX.
 - USB Host and Device stacks are available.
- MK60N512-based TWR-K60N512 Board Support Package was added
 - Standard set of peripheral drivers is enabled GPIO, I2C, INT_CTRL, SERIAL, SPI, SDHC, SD card and FlashX.
 - USB Host and Device stacks are available.
 - Ethernet interface is supported by Enet driver and RTCS TCP/IP stack.
- Changes in the generic PSP code
 - Prototypes of functions which may be in conflict between the MQX RTOS and compiler-specific <string.h> header files were made conditionally compiled by the MQX_SUPPRESS_STRINGH_MACROS configuration option. This suppression is enabled by default for IAR EWARM compiler version 6.
 - New memory allocation API function defined. The _mem_alloc_at and _lwmem_alloc_at functions enable to allocate dynamic memory block at specific address within the memory pool. This is used to implement memory “barriers” between physical RAM blocks which cannot be accessed as a continuous area. (this is true for Kinetis platform).
 - Support for MQX RTOS exception handling implemented for sparse interrupt tables.
 - Kernel log entries added for new memory functions and low-level ENET driver functions.
- Changes in the I/O drivers code
 - New MAC-NET Ethernet module driver with hardware time-stamping support was added to support Kinetis platform.
 - RTC driver API enhanced to be able to work with MQX DATE_STRUCT data type directly.
 - Minor changes done in ADC and GPIO drivers towards generalizing the code across multiple platforms. No functional change.

- Changes in the USB code
 - KHCI Host code significantly rewritten to support new Register I/O maps unified across ColdFire and Kinetis families. No functional change.
- The MQX RTOS documentation was updated:
 - Reference manual was extended by description of `_lwevent_get_signalled()` and `_task_create_at()` API functions.
 - Description of interrupt-level task priorities and NMI handling were extended in the User Manual.

Bug Fixes:

- In the PSP code
 - The `_task_get_template_ptr()` and `_task_get_td()` now safely returns NULL for invalid input parameter values.
- In the RTCS code
 - Compressed domain names haven't been handled correctly in the DNS client parser. Some DNS server responses were not handled properly.
 - Memory leak in the DHCP functionality fixed. Memory was not de-allocated properly during address renewal process.
 - DHCP client requests were not sent complete. With some servers the client was not obtaining the subnet information.
 - UDP service fixed in order to receive frames to a proper socket if the same UDP port is open by multiple network interfaces.
- In the I/O drivers code
 - ADC examples were modified to work properly also on MCF51MM, MCF51AC and MCF51CN platforms.
 - MCF52259 GPIO_PTCPAR_PTCPA macro values were fixed.
 - KBI Interrupt functionality was fixed in the MCF51MM GPIO driver.
 - IO_IOCTL_I2C_GET_BUS_AVAILABILITY command implemented in I2C driver for ColdFire V3 family.
 - IO_IOCTL_SERIAL_SET_DATA_BITS command behavior fixed in serial for ColdFire V1 family.
 - Slave functionality was fixed in DSPI driver for ColdFire families. The slave was sending additional byte after normal data block.
- Other
 - Shell “help” command was fixed to display detailed description properly.

11 Version 3.6.1 (Aug 2010)

- MCF51MM-based TWR-MCF51MM Board Support Package was added
 - Standard set of peripheral drivers is enabled: FLASHX, GPIO, I2C, INT_CTRL, UART, SPI, RTC, CF card, ADC, DAC, SD card

- New driver was added: DAC (API driver).
- USB Host and Device stacks are available. The host (KHCI) driver can be configured to include software workaround for early silicon issue of the MCF51MM device. This workaround is disabled by default.
- MCF53015-based M53015EVB Board Support Package was added
 - Standard set of peripheral drivers is enabled FLASHX, GPIO, I2C, INT_CTRL, SERIAL, SPI, SDHC, CF card, SD card.
 - USB Host and Device stacks are available.
 - Two Ethernet interfaces are enabled in RTCS TCP/IP.
- New SDHC I/O driver covering the eSDHC peripheral module was created. This driver allows low-level access to various cards such as SD, SDHC, SDIO, SDCOMBO, SDHCCOMBO, MMC and CE-ATA. MQX RTOS currently implements upper layer for SD and SDHC card drivers only.
- HTTP server listen address can now be configured in application and may be optionally set to other value than INADDR_ANY. This may help to run the web-server application on just one of several network interfaces.
- New TAD “Memory Blocks Summary” screen implemented for lightweight memory blocks. The screen shows memory allocation grouped by categories.
- The MQX RTOS documentation was updated:
 - MFS User Guide chapter 3.1.9 was updated (search attributes).
 - The lwevent description was updated in MQX RTOS Reference Manual.
 - The description of _time_diff_xxx() function parameters corrected in MQX RTOS Reference Manual.
 - The _RTCS_mem_pool setting documented in chapter 2.6 of RTCS User Guide.
 - IO Guide SPI driver description was extended by new IOCTL command.
- Release notes and Getting started documents were reorganized and extended by more detailed description of the MQX RTOS build process.
- New communication mode was implemented in SPI driver for the QSPI module. In this mode, the communication is split to smaller chunks (16 bytes) to fully leverage hardware-driven CS signal activation. Also in this mode, the CSIV bit is not controlled by the driver flush() function which is normally causing glitch on the other QSPI CS signals.

Bugfixes

- The MFS functionality IO_IOCTL_FIND_FIRST_FILE was not working. If the search attribute MFS_SEAR v2CH_NORMAL was used, no files were found. Attribute matching algorithm was corrected and user manual was updated.
- MFS file write bug which appeared in MQX RTOS 3.6.0 append-file performance optimization was fixed. If two or more files were opened simultaneously, the wrong content can be written as the sector cache was not updated correctly.
- The interrupt enabling code in the GPIO driver was fixed for MCF5223x and MCF5225x devices. GPIO interrupt callbacks now work properly on these platforms.

- When the `_task_restart()` function was restarting the calling task itself, destroying task stack frame can crash the system. The behavior was fixed.
- Touch screen driver did not work properly for M5225X platforms and internal flash targets built in “Release” configuration. The driver code was fixed.
- Setting I2C baudrate did not selected the best closest achievable rate. Algorithm for baud rate calculation was changed to find closest match instead of closest-lower rate as it was before.
- FlashX driver fixed. The bug was causing writing incorrect data if the `fseek()` was used to navigate to the last byte of the Flash file and user wanted to overwrite it.
- Precompiled TWR-MPC5125 libraries were not included in MQX RTOS 3.6.0 release. The libraries are now part of the MQX RTOS 3.6.1 full installation package.
- New MQX RTOS projects created for MPC5125 platform were not using the `MQX_PATH` variable in the access paths. This might cause mismatch in library and include search paths in the newly created projects. This option is now set properly in project stationery.
- USB DATA0/1 bit toggling fixed in the EHCI USB Host driver code. This issue was affecting the PHDC and CDC data transfers.
- USB Device example applications for high-speed data access (EHCI) were updated to use internal data buffers in fast on-chip SRAM memory. This is a temporary solution which will be replaced by generic support of high-speed data transfer in the future.
- The internal bind functionality was fixed in RTCS in order to enable the same TCP port being opened on two different network interfaces.
- Task suspended/running indication bug fixed in TAD. Call history is now displayed correctly for all tasks in both classic CodeWarrior and CodeWarrior version 10.
- Default PLL system clocks setup was changed back to MQX RTOS 3.5.0 values in M52277EVB BSP startup code. The SD RAM chip setting (160MHz) was unstable and caused sporadic errors during debugging.
- The `lwtimer_create_periodic_queue()` functionality was fixed. Timer callback was not invoked in some use cases.

12 Version 3.6.0 (Jun 2010)

- The support for older CodeWarrior for ColdFire 7.1 and CodeWarrior for Microcontrollers 6.2 was removed. Build projects are available only for versions 7.2 and 6.3.
- Eclipse-based CodeWarrior 10 support is added. All library and application build projects are also available in the native CW10 format.
- MCF51AC-based DEMOAC Board Support Package added with basic set of peripheral drivers: ADC, FLASHX, GPIO, I2C, INT_CTRL, SERIAL, SPI.
- MCF51JM-based EVB51JM128 Board Support Package added with basic set of peripheral drivers: ADC, FLASHX, GPIO, I2C, INT_CTRL, SERIAL, SPI. A support for USB Host/Device stack is also available.
- MPC5125 BSP added. This is the first PowerPC port supported by Freescale MQX RTOS. This version includes the code released recently as MPC5125 patch to MQX RTOS 3.5.1.

- Basic set of I/O drivers supporting the MPC5125 peripherals is included: PSC UART, Fast Ethernet controller, DIU, msCAN and RTC.
- Example applications available: Standard set of MQX RTOS and RTCS examples, the “Digital Sign” demo application demonstrating the graphical DIU module of MPC5125 and the HDMI interface.
- The `MQX_SUPPRESS_FILE_DEF` and `MQX_SUPPRESS_STDIO_MACROS` may now be defined as 1 in `user_config.h` to avoid declaration of generic `FILE`, `FILE_PTR` types and `<stdio.h>` functions like `read`, `write`, `printf` etc.
 - The MQX RTOS code was refactored to use the new `MQX_FILE` and `MQX_FILE_PTR` types and make aliases to backward compatible `FILE` and `FILE_PTR` types only if not suppressed.
 - The key changes were implemented in the file `fio.h` in `mqx/source/include`.
 - Default values of both configuration options are defined such that backward compatibility is not affected.
- The `user_config.h` files and shared configuration files in `/config/common` directory were re-formatted and updated:
 - All `BSPCFG` options related to given board are now available in individual `user_config.h` files.
 - `MQX_TASK_DESTRUCTION` and `MQX_EXIT_ENABLED` options are now set to 0 in `small_ram_config.h` file.
 - `MQX_COMPONENT_DESTRUCTION` option was removed from the `verif_enabled_config.h` file. The default value of 1 defined in `mqx_cnfg.h` is used. The ColdFire V1 processors override this option to 0 in `user_config.h` to save code size.
- A new `_lwevent_get_signalled()` API call was added to enable detecting which lwevent bits caused the active task to return from `_lwevent_wait()` call.
- Inline assembler code was eliminated where possible to simplify migration to alternative C compilers.
 - MCF52xx and MCF51xx FlashX flashing routines were rewritten in C.
 - The “nop” instruction primitive `_ASM_NOP()` added to compiler-specific header file and used instead of `asm(nop)` inline assembler statement.
 - The “stop” instruction primitive `_ASM_STOP(x)` added to compiler-specific header file and used instead of `asm(stop #(x))` inline assembler statement.
- Support for Freescale eGUI software and TWR-LCD display board has been added. See eGUI MQX RTOS demos within the Freescale eGUI package. The eGUI comes with set of low-level drivers for non-OS bare-metal operation. The drivers in MQX RTOS 3.6 enable the eGUI package to be configured for hardware-independent “MQX RTOS” mode where all board-dependent code resides in the MQX RTOS BSP.
 - Resistive Touch Screen driver added to TWR-MCF52259 and TWR-MCF51CN BSPs.
 - LCD memory regions added to TWR-MCF52259 and TWR-MCF51CN linker command files in BSPs.
 - GPIO pins, ADC channels and SPI channel-specific for LCD and touch screen control are defined in TWR-MCF52259 and TWR-MCF51CN BSP header files.

- Inter-processor communication (IPC) over the UDP protocol was added together with an example application in RTCS.
- RTCS ipconfig shell command enhanced to be able to print immediate values of low-level PHY registers.
- Serial driver supports new open mode-flag (`IO_SERIAL_NON_BLOCKING`) to enable accessing the serial driver in non-blocking mode. Supported by both polled and interrupt serial drivers.
- New I2C ioctl command (`IO_IOCTL_I2C_GET_BUS_AVAILABILITY`) implemented to detect idle/busy condition on the bus.
- Register I/O map structures were updated to cover all supported parts and changes in I/O drivers. In addition to changes in processor-specific header files, the following shared I/O map files have been changed:
 - `mcf5xxx_spi16.h` ... SPI16 no-FIFO module added.
 - `mcf5xxx_usb0tg.h` ... ULPI control registers added.
 - `mcf522x_usb0tg.h` ... registers added
 - `mcf51xx_i2c.h` ... I2C with/without SMB support defined
 - `mcf51xx_ftsr.h` ... registers added
 - `MQX_COMPONENT_DESTRUCTION` option was removed from the `verif_enabled_config.h` file. The default value of 1 defined in `mqx_cnfg.h` is used. The ColdFire V1 processors override this option to 0 in `user_config.h` to save code size.
- A new `_lwevent_get_signalled()` API call was added to enable detecting which lwevent bits caused the active task to return from `_lwevent_wait()` call.
- Inline assembler code was eliminated where possible to simplify migration to alternative C compilers.
 - MCF52xx and MCF51xx FlashX flashing routines were rewritten in C.
 - The “nop” instruction primitive `_ASM_NOP()` added to compiler-specific header file and used instead of `asm(nop)` inline assembler statement.
 - The “stop” instruction primitive `_ASM_STOP(x)` added to compiler-specific header file and used instead of `asm(stop #(x))` inline assembler statement.
- Support for Freescale eGUI software and TWR-LCD display board has been added. See eGUI MQX RTOS demos within the Freescale eGUI package. The eGUI comes with set of low-level drivers for non-OS bare-metal operation. The drivers in MQX RTOS 3.6 enable the eGUI package to be configured for hardware-independent “MQX RTOS” mode where all board-dependent code resides in the MQX RTOS BSP.
 - Resistive Touch Screen driver added to TWR-MCF52259 and TWR-MCF51CN BSPs.
 - LCD memory regions added to TWR-MCF52259 and TWR-MCF51CN linker command files in BSPs.
 - GPIO pins, ADC channels and SPI channel-specific for LCD and touch screen control are defined in TWR-MCF52259 and TWR-MCF51CN BSP header files.
- Inter-processor communication (IPC) over the UDP protocol was added together with an example application in RTCS.

- RTCS ipconfig shell command enhanced to be able to print immediate values of low-level PHY registers.
- Serial driver supports new open mode-flag (IO_SERIAL_NON_BLOCKING) to enable accessing the serial driver in non-blocking mode. Supported by both polled and interrupt serial drivers.
- New I2C ioctl command (IO_IOCTL_I2C_GET_BUS_AVAILABILITY) implemented to detect idle/busy condition on the bus.
- Register I/O map structures were updated to cover all supported parts and changes in I/O drivers. In addition to changes in processor-specific header files, the following shared I/O map files have been changed:
 - mcf5xxx_spi16.h ... SPI16 no-FIFO module added.
 - mcf5xxx_usbotg.h ... ULPI control registers added.
 - mcf522x_usbotg.h ... registers added
 - mcf51xx_i2c.h ... I2C with/without SMB support defined
 - mcf51xx_ftsr.h ... registers added
- Register I/O map structures of MCF54455 were expanded to cover peripherals not supported by MQX RTOS drivers (DMA timers, EPORT, PCI, RNG, SSI).
- The entire code was “untabified” (TAB characters replaced by spaces).
- Readme files created for selected example applications (GPIO, I2C, etc.).
- IPC demo application code generalized for various TTY ports availability.
- Codesize MAP-file analysis script now prints code-size statistics for RTCS, MFS and other system libraries.
- MQX RTOS I/O User Guide updated:
 - SPI, I2C, ADC, RTC and SD Card driver sections updated.
 - ESDHC, FlexCAN, DAC Driver and NAND Flash Driver sections added.
- Board-specific information related to MQX RTOS release has been moved from Release Notes into a separate document (see *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS)).

Bugfixes in 3.6.0

- MQX RTOS Kernel:
 - The _io_fclose() no longer locks the kernels IO_LWSEM semaphore before calling the device drivers close function. This can cause a race condition with layered drivers.
 - The EWL library linking is now disabled in all MQX RTOS library projects (PSP, BSP, MFS, RTCS, ...). Having this option enabled in “Librarian” projects settings panel was causing all EWL code to be included in each MQX RTOS library. This was effectively disabling EWL function overrides from MQX RTOS code. For example the malloc() code was taken from EWL instead of from MQX RTOS and was causing linking issues (malloc is also part of C++ new operator implementation).
 - Handling of priority messages was fixed. In some situations, a priority message could have been received from a wrong message queue.

- Format specifier parameter is declared as const in the sscanf prototype. This avoids compiler warning messages when passing constant string as the scan format.
- The MQX RTOS watchdog component (do not confuse this with the on-chip hardware watchdog) was fixed in _watchdog_isr() to expire at proper time on Coldfire V1 platforms. Potentially this also solves problems if the Watchdog uses other timer than the main BSP tick timer.
- BSPs and I/O drivers:
 - EPORT interrupt level macros in MCF522xx board-specific headers in BSP are now set correctly according to the levels hardwired in the controller device. Be aware that EPORT_EPF7 level is hardwired to non-maskable interrupt so no MQX RTOS calls may occur in the interrupt routine.
 - GPIO button pin identifiers (BSP_SW1 and BSP_SW2) were fixed in M52233DEMO BSP header.
 - ADC driver functions now return more detailed error codes instead of generic IO_ERROR value.
 - MCF522xx interrupt initialization and unmasking functions fixed. Problem was that the code was also inadvertently unmasking other peripheral interrupt sources in the interrupt controller.
 - TWR-MCF51CN tick timer interrupt handler was fixed in BSP. Due to wrong timer interrupt acknowledge code, the MQX RTOS system tick time was one half of the configured value.
 - MCF51EM-specific code was removed from generic MCF5xxx RTC driver code.
 - I2C clock source changed from BSP_BUS_CLOCK to BSP_SYSTEM_CLOCK on all MCF52259-based boards. This fixes the problem of I2C bitrate which was 2x higher than configured.
 - Uncached memory region definitions were fixed in M54455EVB linker command files in BSP.
 - Default PLL system clocks setup was changed in M52277EVB BSP startup code. The MCF52277 processor now runs at system clock 160MHz (80MHz bus clock, 60MHz USB clock).
 - USB I/O pin initialization routine was fixed in M52277EVB BSP. The problem with pin settings made USB Host functionality unusable on M52277EVB.
 - SPI chip select control optimized for performance.
 - MCF5xxx FEC driver initialization of 2nd Ethernet interface (ipconfig 1 init) is no longer dependent on having 1st interface initialized.
 - Compiling MCF5xxx FEC code generates error message if un-cached memory support is disabled in MQX RTOS configuration.
- MFS Filesystem:
 - The CLUSTER_SIZE_BYTES member of MFS_DRIVE_STRUCT is now extended to 32 bit to avoid size overflow for 65 k cluster size (solves some USB memory stick problems).
 - Hidden attribute checking was fixed in MFS_Attribute_match() function.
 - MFS write/append performance problem was solved by avoiding unnecessary sector reading.
- SHELL Library:
 - Memory violation error fixed in the shell “help” command implementation.

- USB Host Stack:
 - USB Host HUB class enhanced to support asynchronous status inquiries.
 - USB CDC Host driver now waits for control transactions to ACM interface to complete. This fixes a problem of invalid memory access with some CDC devices.
 - KHCI USB bus control is deactivated after device is detached. This avoids problems where noise on detached bus was evaluated as a valid communication.
 - KHCI USB data bus weak pull-down resistors enabled to improve detection of reset/detach event.
 - EHCI fixed to toggle data tokens correctly when a transfer has odd number of transactions.
- Example applications:
 - Digital sign demo running on TWR-MPC5125 now runs the display task with higher priority than the network task. This fixes a problem where images were displayed after 3 minutes ARP timeout when board was not connected to network.
 - Bar-graph CGI code in HTTP Server demo now reads ADC channels properly.
 - msCAN example application fixed on TWR-MPC5125.
- Task Aware Debugger plug-in for CodeWarrior
 - The value of SR register is now displayed correctly in context of any tasks being examined.
- Other:
 - New Version Checker tool was fixed so it always runs in a single instance. Each of the TAD DLLs (current version and all backward-compatibility versions) starts an instance of the web checker tool upon CodeWarrior start. Running the tool in multiple instances was causing problems with loading the webchk.wcp configuration file or resulted in multiple “New Version Available” notifications being displayed.

13 Version 3.5.1 (Feb 2010)

- Fixed bugs:
 - MCF5225x ADC driver - pin initialization was not performed properly; problem appears with “Release” targets.
 - FlashX driver was not working correctly with internal Flash memory of MCF52xx processors.
- Example applications updated
 - MFS SD Card example application demonstrates use of Partition Manager now.
 - Code-size benchmark applications and results are available for M52259EVB, M54455EVB, TWR-MCF51CN.

14 Version 3.5.0 (Jan 2010)

- CodeWarrior build projects location was changed. The build/codewarrior subdirectory was replaced by several directories depending on type and version of the CodeWarrior tool.
 - build/cwmcu62 contains build projects for CodeWarrior Development Studio for Microcontrollers version 6.2 (ColdFire V1 projects from MQX RTOS 3.4 and earlier)

- build/cwmcu63 contains build projects for CodeWarrior Development Studio for Microcontrollers version 6.3 (New versions of ColdFire V1 projects)
- build/cwcf71 contains build projects for CodeWarrior Development Studio for ColdFire Architectures version 7.1 (ColdFire V2-V4 projects from MQX RTOS 3.4 and earlier)
- build/cwcf72 now contains build projects for CodeWarrior Development Studio for ColdFire Architectures version 7.2 (New versions of ColdFire V2-V4 projects)
- The post-linker batch files were separated to build/bat out of the build folders to be able to reuse them between different build project versions.
- Mass-build projects of all libraries were also moved from config/<board> folder to an appropriate subdirectory.
- Build projects for the later CodeWarrior versions (cwcf72 and cwmcu63) no longer support standard (on-stack) parameter passing. StdABI targets are not available in build projects.
- The code and folder structure was updated to make it ready for IAR toolset support
 - CodeWarrior-specific C and Assembler syntax was changed to be compatible with IAR compilers. Only few parts of code made conditionally compiled depending on the compiler.
 - CodeWarrior-specific portions of BSP (start-up code, memory initialization code, debugger configuration files) were moved to cw subfolder. The iar subfolder will be added by the IAR support patch in the future.
- Changes in PSP
 - CPU numbering scheme was changed. The CPU is still defined in the user_config.h file, but uses constants pre-defined in psp_cpudef.h file.
 - Several new code-size compile-time configuration options were added. See updated *MQX RTOS User's Guide* (document MQXUG) documentation.
- Changes in MFS
 - Dynamic read-only status checking of the physical device was added to MFS write functions as optional feature (MFSCFG_READ_ONLY_CHECK). This option is useful for example with SD Card storage with write-protect switch.
- RTCS stack updated
 - Set of RTCS iwcfg_XXX() functions was implemented to enable application-level control of any future WiFi functionality.
 - ARP resend and expiration timeout values were made configurable through the user_config.h file. Especially the “expire incomplete” time needs attention as it affects the total time a send call takes when target IP address does not exist on the local subnet.
 - Default setting for DHCP DISCOVER message was changed. Broadcast flag is set to FALSE to request unicast DHCP responses by default.
 - Internal structure names were renamed to keep consistency with C typedef names. The structure tag names are now always the lowercase variant of the C typedef name. This change enables proper ELF file format parsing by Task Aware Debugger plugin.
 - Bug fixed – RTCS_selectset function now behaves as in RTCS versions prior to 3.0. The socket array is not altered by this function.
 - RTCS_select call enhanced to support more than one socket in the array.

- Shell library was extended by new “iwconfig” command to enable the iwcfgf_xxx() control from the Shell environment
- BSP and support for MCF51EM256-based DEMOEM board was added
 - SCI, SPI, I2C and other standard drivers were ported to this platform
 - ADC, RTC, FlashX were enhanced to support additional features the MCF51EM offers
- ENET Driver was updated
 - Generic Wi-Fi support was added into driver code for future WiFi support.
 - ENET MAC interface structure was extended to support both generic and device-specific control commands (so-called media control commands).
 - Set of Wi-Fi-related media control commands was defined (setting the ESSID, encryption, ...)
 - ENET driver initialization structure was extended by new device-dependent initialization parameter.
 - ENET driver concept of device context locking was enhanced (from global interrupt disable to lwsem-based locking).
 - HVAC, WEB_HVAC, HTTPSRV and RTCS Shell example applications were made ready for WiFi functionality.
- SPI driver was updated
 - Support for SPI16 module of MCF51EM was added.
 - Driver code for 8-bit ColdFire V1 SPI module was internally renamed to SPI8.
 - The SPI driver now provides uniform I/O API to SPI8, SPI16, QSPI and DSPI modules.
 - Bug fixed – QSPI chip select was incorrectly asserted during driver initialization making a glitch on this signal.
- GPIO driver was extended
 - Open flags were added to control polarity of edge which triggers interrupt callback routine to be called (rising/falling edge).
 - I/O control commands were added to enable/disable interrupt callback function to be called.
- ADC driver now supports the PDB trigger module and sixteen-bit ADC module available at MCF51EM.
 - ADC driver install function was changed to accept additional device-specific initialization structure.
- SD Card driver now supports the SD Card interface available on M52277EVB. SD Card Example application was added for this board.
- Example applications
 - SD Card example application now decodes GPIO inputs for card presence and read-only switch.
 - Code-size benchmark application was added for MCF52259EVB. The application makes artificial reference to MQX RTOS kernel API functions so the kernel code-size can be analyzed from map files. A tool automating the build, map file parsing, and HTML report generation is also available.

- Timing benchmark application was added for MCF51CN, MCF52259 and MCF54455 devices. This application measures key timing parameters of the MQX RTOS kernel. HTML report generator tool is also available.
- RTC example application enhanced to support new features available in MCF51EM.
- SPI example application extended from EEPROM testing to general SPI-based memory device (EEPROM, Flash or serial MRAM).

15 Version 3.4.0 (Sep 2009)

- Support for Register ABI (register parameter passing) was implemented
 - RegABI build targets (for both Release and Debug configurations) added in all library build projects. The binary libraries compiled with RegABI configuration get the “_regabi” postfix.
 - Former build targets which use the Standard ABI are still maintained in the library build projects for backward compatibility. The targets were renamed to “StdABI”.
 - All example and stationery applications were reconfigured to use Register ABI and RegABI MQX RTOS libraries.
- M52277EVB BSP added
- M52233DEMO BSP added
- USB EHCI Host Support implemented
 - USB Host functionality enabled on MCF52277 and MCF54455.
- USB Device Stack reworked and enhanced
 - The USB Device Stack code has been partially rewritten to be consistent with the similar bare-metal stack available for Freescale HCS08 platform (released separately).
 - CDC class implementation examples added (virtual serial line and virtual network interface card).
 - PHDC medical class implementation examples added.
- SPI driver reworked to support all kinds of ColdFire SPI modules (SPI, QSPI and DSPI)
 - Former QSPI driver was removed from the release.
- IPC inter-processor communication files made available in the release.
 - IPC Example applications provided.
 - SCI-based IPC tested only.
- SPI-based SD Card Driver added
 - The driver was tested with Memory storey for Tower Kit only (MCF51CN and MCF52259 devices). Not tested with M52277EVB board.
 - Bug fixed: Wrong SCI baud rate divisors calculation fixed.
 - Bug fixed: Shell “dir” command file attribute filter is now applied correctly.
 - Bug fixed: MFS read and write calls correctly return negative value when physical device returns access error. Please see also another known issue described in section .

16 Version 3.3.0 (Aug 2009)

- Added BSP for MCF52259 Tower Kit.
 - CodeWarrior projects, demos and example applications were ported from M52259EVB to this BSP.
 - CodeWarrior projects and the BSP also support debugging targets and running code from external MRAM memory. This is prepared for future support of Memory Storey board for the Tower Kit.
- Added PSP, BSPs and other support files for M54455EVB ColdFire V4 system. This is the first V4 device supported by Freescale MQX RTOS operating system. Few new features were added to PSP:
 - Cache control support added. Non-cached memory pool allocation API added.
 - Flash-to-RAM code copying enabled in startup.
 - Startup code is now part of PSP. Two files (CF_startup.c and ROMCopy.c) were reused from CodeWarrior runtime library. These files were modified to support ROM-to-RAM code copying.
 - Flash Programmer config files available in tools/flash_programmer_scripts/ config directory.
 - SPI driver for M54455EVB created (the DSPI module), example supports onboard SPI Flash
- Ethernet driver and RTCS were significantly re-written.
 - Device-independent part of the driver was re-written to support multiple Ethernet MAC devices of the same or even different kinds. A support for memory-optimized handling of small frames was also added.
 - Device-dependent part of the Ethernet driver was re-written to support ColdFire V1-V4 Fast Ethernet Controller module.
 - Several RTCS features were re-tested and fixed to support multiple physical controllers.
- RTCS updated
 - The rtcs_shell example now uses the MFS lib (with RAM-disk) and supports all FTP commands (conditional compilation)
 - Minor changes in IPCFG API to support multiple devices, potentially of different kind than ENET.
- TAD updated
 - Strings displayed in TAD moved to separate text files available inside MQX RTOS installation.
 - New Ethernet driver screen added.
 - Symbolic reader enhanced some objects (LWsem, LWevent, ...) are displayed not only with address information, but also with symbolic names.
- Several key TAD-like screen dumps implemented also on embedded application side in PSP. These functions may be used to print out the TAD-like debugging data by the application.
- Shell library updated
 - RTCS support commands changed (ipconfig, netstat).
 - MFS format command was made available (also see mfs/examples/ramdisk application)

- Shell interface to TAD-like screens dumps added.
- Workaround for MCF5223x Ethernet PHY auto-negotiation issue implemented according to chip errata.
- FlashX driver updated:
 - Support of external Flash devices made available. Tested with external flash memory devices of M54455EVB.
 - FlashX example application modified to support external flash1 device on M54455EVB.
 - Write protect ioctl command implemented for external Flash devices.
- USB Host updated:
 - Code refactored and generally updated (MASS storage class renamed to MSD).
 - CDC class added. Example application shows how to forward characters between standard UART TTY and a virtual USB TTY device.
- I2C driver refactored and updated, eeprom example changed.
- CodeWarrior “Stationery” templates for creating new MQX RTOS projects were changed:
 - All kinds of stationery projects now have full set of libraries added to project (BSP, PSP, MFS, RTCS, USB Host, USB Device, Shell). The standard CodeWarrior linker will optimize out the unreferenced code. Feel free to remove the unused libraries from the project.
 - MQX-Only stationery is a simple “hello-world” like example.
 - MQX+MFS stationery is simple shell example working with RAM disk.
 - MQX+RTCS stationery is simple shell example with basic networking commands and telnet server.
 - MQX+RTCS+MFS stationery is a union of the two examples above.
 - USB Host is a simple application able to detect devices attached to the host port or devices attached through USB hub. You still need to see the USB host example applications to see how to interface attached devices.
 - The USB Device stationery is not available in this release. Please see the USB Device example applications.
- Device-specific sections added to this release notes document. Please read the required jumper settings for each evaluation board supported.

17 Version 3.2.1 (May 2009)

- Sparse interrupt table implemented and made a default option for small-RAM devices. This feature saves up to 1k of RAM. The TAD CodeWarrior plug-in was updated to support this feature.
- The legacy MQX RTOS Flash driver (called FlashX) was ported to support internal Flash memory of all supported devices (MCF51CNxxx, MCF522xx). The Flash driver requires Flash parameters to be passed from the application or linker command file. The linker command files were updated in all BSP. Sample application is provided.
- The GPIO driver callback-on-interrupt feature was ported to MCF52xx devices. This feature is still a subject to change in the future versions. The MCF51xx driver was optimized for code size.
- Minor bug fixes in Serial driver (fflush system call implemented).

- MCF51CNxx SPI example enhanced to support both SPI EEPROM and SPI Flash devices. The example applications may be used with Memory Storey board.
- Fixed known issue with MQX RTOS Stationery as it was described in the last release notes. The MQX_PATH named source tree is no longer used for referencing the debugger initialization file (this was causing Flash Programmer crash). Copies of debugger initialization and memory configuration files were made part of all stationery projects and are referenced using relative path only in the debugger settings.
- I/O Driver User Guide updated. The GPIO, ADC, SPI and FlashX driver description added.

18 Version 3.2.0 (May 2009)

- Added PSP, BSPs and other support files for TWR-MCF51CN-KIT Tower Kit with ColdFire V1 processor MCF51CN128.
 - RTCS library was also ported to MCF51CN128 device.
- Ethernet driver significantly re-written and generalized to cover both ColdFire V1 and V2 processors. Application is now able to change some of the driver parameters dynamically, without need to recompile the driver code.
- Most of the features missing in the previous release of the ADC driver were implemented. The driver documentation is still on the to-do list.
- ADC driver partially re-written and low-level layer ported to ADC module of MCF51CN device. There are implementation differences between MCF51CNxx and MCF522xx drivers which are still to be addressed by future releases.
 - MCF51CN driver uses internal lightweight timer component
 - MCF522xx driver still uses PIT timer for internal timing. Implementation with lightweight timer (lower performance) is planned for future versions
- New GPIO callback-on-interrupt feature was added to GPIO driver of MCF51CN. This feature will be ported to other supported devices in the future releases.
- Three new “security” demo applications were created for MCF51CN device demonstrating the ADC and GPIO drivers, network protocols and low-power mode of operation. The applications will be ported to other devices in the future.

19 Version 3.1.0 (April 2009)

- USB Device low-level driver has been implemented for MCF522xx family and example applications were created for M52259EVB and M52223EVB.
- The “usb” subdirectory of the /lib output folder was split to “host” and “device” parts.
- Added PSP, BSPs, and other support files for MCF52223 and MCF52235 evaluation boards:
- USB Host and Device libraries were ported to M52223EVB
- RTCS library was ported to M52235EVB
- MFS and Shell libraries ported to both new boards
- USB Host HUB issue resolved with MCF52259 (excessive number of errors observed when USB devices were accessed through USB HUB). The issue was solved by implementation of SOF frame

scheduler in the USB Host low-level driver. An issue with a similar impact remains on MCF5222x implementation (silicon problem, no workaround known).

- Error codes naming and numbering convention has been made more consistent across MQX RTOS, RTCS, and other libraries. Old error code names remain implemented for backward compatibility (numeric values changed).
- Kernel Data and other internal structures were optimized for size. Parts of the structures were made conditionally compiled based on the user configuration.
- TAD (CodeWarrior debugger plugin) was updated to handle renaming in internal structures.
- Memory type information was also added to lightweight memory structures. TAD is now able to display the type of information for memory blocks allocated by using lightweight memory allocator.
- Lightweight memory allocator has been made the default option for all BSPs.
- The /config/<board>/user_config.cw files were eliminated. Such files have contained a subset of configuration options for assembler-coded files in the MQX RTOS kernel. All kernel assembler-coded files are now preprocessed using C preprocessor and make use of standard C header macros. The C-language-specific content of the header files is conditionally excluded during assembler compilation by using `__ASM__` macro.
- The assembler vectors.cw file in each BSP was re-coded to C syntax. A new user configuration macro (MQX_ROM_VECTORS) may be set non-zero to avoid vector table being copied to RAM.
- The BSP, PSP, RTCS and other library build projects were changed to be easier to understand (virtual folders inside each CodeWarrior project was updated).
- SNMPv2 code was changed to enable ROM-based MIB structures. Example application is provided demonstrating custom MIB nodes and user traps.
- Several source code files were renamed to better reflect the content, especially in the I/O driver directories.
- Added “root directory” concept to the FTP server, “rename” command added.
- SPI I/O driver added (Master mode only). Example application is available.
- Initial ADC I/O driver was added. This driver (including its API) is subject to improvement in the future releases. ADC device driver usage is demonstrated in ADC example application.
- RTC API updated, example application added.
- The new IPCFG API updated and documented
- The new HTTP server API documented
- MQX RTOS I/O Driver User Guide added

20 Version 3.0.1 (Jan 2009)

- Small enhancements through the whole code base of MQX RTOS kernel and other components
 - Memory block “type” information added to all system memory allocations. TAD is now able to give detailed information about each memory block allocated by kernel or system component.

- Dedicated memory allocation routines in RTCS, MFS, and USB simplify the memory pool usage.
- IPCFG Ethernet link monitoring features and automatic IP address binding functionality was added to RTCS. The “ipconfig” shell command replaces the old “bind”, “ifbind” and “dhcp” shell commands.
- The HTTP server in RTCS was re-written to enable multiple sessions to be served by a single task. This feature brings more reliable HTTP server while maintaining low memory requirements.
- USB Host HUB class was added.
- MQX RTOS User Guide and several other documents are included in the setup package.

21 Version 3.0.0 (Dec 2008)

- This is initial release supporting the MCF52259 processor, M52259EVB, and M52259DEMO boards.

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, CodeWarrior, ColdFire, Vybrid, and Kinetis are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Tower is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. ARM, ARM Powered logo, and Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.

© 2015 Freescale Semiconductor, Inc.

Document Number: MQXCHLGUG
Rev. 0
06/2015

